

St. John's Law Review

Volume 66
Number 4 *Volume 66, Fall-Winter 1993, Number*
4

Article 7

April 2012

The Second Circuit's Attempt to Define Copyright Protection for Computer Software: Is the Abstraction-Filtration-Comparison Test a Workable Solution?

Martin T. Hillery

Follow this and additional works at: <https://scholarship.law.stjohns.edu/lawreview>

Recommended Citation

Hillery, Martin T. (1993) "The Second Circuit's Attempt to Define Copyright Protection for Computer Software: Is the Abstraction-Filtration-Comparison Test a Workable Solution?," *St. John's Law Review*. Vol. 66 : No. 4 , Article 7.

Available at: <https://scholarship.law.stjohns.edu/lawreview/vol66/iss4/7>

This Note is brought to you for free and open access by the Journals at St. John's Law Scholarship Repository. It has been accepted for inclusion in St. John's Law Review by an authorized editor of St. John's Law Scholarship Repository. For more information, please contact seljbyc@stjohns.edu.

THE SECOND CIRCUIT'S ATTEMPT TO DEFINE COPYRIGHT PROTECTION FOR COMPUTER SOFTWARE: IS THE ABSTRACTION-FILTRATION- COMPARISON TEST A WORKABLE SOLUTION?

The tremendous growth in the personal computer industry during the past decade¹ has spurred rapid development in the software market.² Once a highly technical and expensive product available only to commercial users, software³ may now be purchased by individuals at a reasonable price.⁴ However, because

1. See STATISTICAL ABSTRACT OF THE UNITED STATES 761 (1991) [hereinafter STATISTICAL ABSTRACT]. In 1981, 1.1 million personal computers were purchased at a cost of \$3.1 billion as contrasted with 6.75 million units sold in 1985 for \$17.1 billion. *Id.* By 1988, sales reached 9.5 million totalling \$27.7 billion in revenues. *Id.* Moreover, an estimated 23% of the households in the United States owned personal computers in 1989, as compared to 13% percent in 1985. See Trish Hall, *Electronics: It's Not Home Without It*, N.Y. TIMES, Mar. 29, 1990, at C6.

2. See Howard Root, Note, *Copyright Infringement of Computer Programs: A Modification of the Substantial Similarity Test*, 68 MINN. L. REV. 1264, 1264 (1984). During the past forty years, microcomputers and mass-marketed software have grown and evolved with "phenomenal speed." *Id.* "An estimated one million computer programs are created each year." *Id.* at n.2 (citation omitted). This results from the fact that "[t]he world is undergoing an information explosion that is causing a demand for new products and services to help manage vast amounts of information efficiently and effectively." Victor Siber, *The Worldwide Status of Software Protection*, NAT'L L.J., Jan. 21, 1985, at 20.

3. See DAVID BENDER, COMPUTER LAW: EVIDENCE & PROCEDURE § 2.06 (1990). Computer experts and the courts have not defined the terms "software" and "computer program" in any uniform manner. *Id.* Conceptually, the two words are the same, but computer programs have been distinguished as being "sets of instructions that operate the computer," R. LEE HAGELSHAW, THE COMPUTER USER'S LEGAL GUIDE 87 (1985), while computer software "is the collection of the materials that contains, expresses and explains a computer program," *id.* at 87. The copyright statute defines a computer program as a "set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101 (1988). In this paper, the terms "software" and "computer program" will be used interchangeably, and the definition of these terms will be limited to the set of materials that instructs the computer. See BENDER, *supra*, § 2.06 (defining software as computer programs).

4. See Thomas L. Hazen, *Contract Principles as a Guide for Protecting Intellectual Property Rights on Computer Software: The Limits of Copyright Protection, The Evolving Concept of Derivative Works, and the Proper Limits of Licensing Arrangements*, 20 U.C. DAVIS L. REV. 105, 105-06 (1986). The consumer-oriented software market has evolved into a

software is relatively easy to copy⁵ and computer literacy is on the rise,⁶ protection against unauthorized copying⁷ has become a primary concern of computer software companies. Unfortunately, the law has increasingly been outpaced by the technology in the computer software field.⁸ Thus, it is often unclear what protection, if any, is afforded to a particular computer program.⁹ Copyright re-

major industry. *Id.* at 106. For example, in 1989, magazine advertising for computers, stationery and office equipment totalled \$288 million as contrasted with only \$78 million in 1980. See STATISTICAL ABSTRACT, *supra* note 1, at 563.

5. See *Apple Computer, Inc. v. Formula Int'l, Inc.*, 562 F. Supp. 775, 783 (C.D. Cal. 1983), *aff'd*, 725 F.2d 521 (9th Cir. 1984). "[T]he process of duplicating or copying a program, once it is put into usable form . . . is almost absurdly simple." *Id.* In addition, "[d]iskettes can be copied for a minimal cost by anyone with rudimentary technical skill." *Id.*; see also *Root, supra* note 2, at 1264 ("pirating" of software is easy and prevalent); Mark M. Friedman, *Copyrighting Machine Language Computer Software—The Case Against*, 9 COMPUTER/L.J. 1, 2 (1989) ("pirating" software has reached epidemic proportions and adversely affects industry).

6. See STATISTICAL ABSTRACT, *supra* note 1, at 150. In 1989, 96% of elementary and secondary level institutions used computers in the curriculum. *Id.* Computer use in the business environment was over 38% in 1989. *Id.* at 412.

7. See Charles Cangialosi, *The Electronic Underground: Computer Piracy and Electronic Bulletin Boards*, 15 RUTGERS COMPUTER & TECH. L.J. 265, 271-73 (1989). It is estimated that in 1988 over \$1 billion in revenue was lost world-wide because of unauthorized copying. *Id.* Due to the increased losses in sales and corresponding loss of production and jobs resulting from the theft of computer software, unauthorized copying must be confronted not only to remedy the immediate injury, but for the long term deterrence. *Id.* at 300-01. In addition, because the actual writing of the software code is relatively easy compared to the amount of work, expense and effort that goes into a program's logic, structure, and maintenance, software deserves protection from piracy. See Friedman, *supra* note 5, at 16. In fact, only 20% of the program's cost is attributable to the coding itself. See *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1231 (3rd Cir. 1986) (citation omitted), *cert. denied*, 479 U.S. 1031 (1987).

8. See *Root, supra* note 2, at 1268-69 ("copyright laws have only recently been interpreted to extend to computer software"). "Congress, as well as the judiciary, is hard pressed to keep current on computer developments." *Id.* at 1269 n.34.

In 1975, Congress created the National Commission on New Technology Uses of Copyrighted Works ("CONTU") to study the problems associated with intellectual property. See FINAL REPORT OF THE NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS 1 (1979) [hereinafter CONTU REPORT]. Prior to receiving CONTU's recommendations, Congress passed the Copyright Act of 1976, which classifies computer programs as "literary works." See 17 U.S.C. §§ 101, 107 (1988). In 1978, CONTU issued its report containing a number of recommendations. See CONTU REPORT, *supra*, at 1-3. One recommendation was to make clear that computer programs, "to the extent that they embody an author's original creation, are proper subject matter of copyright." *Id.* at 2. Congress adopted the recommendations in 1980, and explicitly extended copyright protection to computer software. See 17 U.S.C. §§ 101, 107 (1988). For additional information on copyright protection of software, see *infra* notes 47-95 and accompanying text.

9. See MICHAEL C. GEMIGNANI, *THE LAW AND THE COMPUTER* 80 (1981). Patent law, trade secrecy, and copyright have all been used to protect ideas from copying without permission; however, the evolution of computer technology has presented legal problems that

strictions are vague, and there remains a great deal of confusion as

have not previously been encountered. *Id.* See generally BENDER, *supra* note 3, § 3 (discussing methods available to protect software).

Although copyright protection for computer software has been available since 1964, only 1205 programs were submitted for copyrighted registration between 1964 and 1978, and over three quarters of the software registered were owned by two of the largest hardware manufacturers, IBM and Burroughs Corp. See Mickey T. Mihm, Note, *Software Piracy and the Personal Computer: Is the 1980 Software Copyright Act Effective?*, 4 COMPUTER/L.J. 171, 180 (1983); GEMIGNANI, *supra*, at 86 (noting little interest in copyright of software). The reason for such a low rate of registration was the uncertainty over whether copyright laws would protect a program's underlying ideas and methods. See Raymond J. Areaux, Comment, *Computer Software Protection: From Infancy to Adolescence*, 31 LOY. L. REV. 301, 310-11 (1985).

While copyright protection for programs is relatively inexpensive and easy to obtain, "it is patent protection for which the software industry has lobbied most actively." See GEMIGNANI, *supra*, at 93. Patent protection is available for "any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof." 35 U.S.C. § 101 (1988). It only protects creations that are not obvious. See *Graham v. John Deere Co.*, 383 U.S. 1, 3 (1966). If one product accomplishes substantially the same result as a patented product, in the same manner, then the patent has been infringed, regardless of any difference in form, shape or name. See *Graver Tank & Mfg. Co. v. Linde Air Prods. Co.*, 339 U.S. 605, 608 (1950) (citing *Sanitary Refrigerator Co. v. Winters*, 280 U.S. 30, 42 (1929)). The Supreme Court has withheld patent protection from certain scientific areas, such as laws of nature, see *Eibel Process Co. v. Minnesota & Ontario Paper Co.*, 261 U.S. 45, 52 (1923) (refusing to grant patent protection to principle that water will run downhill), and mathematical expressions, see *MacKay Radio & Tel. Co. v. Radio Corp. of Am.*, 306 U.S. 86, 94-102 (1938) (improving empirical mathematical formula not patentable invention).

The Patent and Trademark Office initially took the view that patent protection was not available for computer programs because they encompassed mathematical processes or expressions. See Nelson Moskowitz, *The Metamorphosis of Software—Related Invention Patentability*, 3 COMPUTER/L.J. 273, 282-83 (1982). CONTU, which was created by Congress to study the implications of the new technologies in intellectual property law, had serious doubts "whether a patent may ever be obtained for a computer program." See CONTU REPORT, *supra* note 8, at 42.

The Supreme Court, however, has clearly indicated that patent protection is available for computer programs. See *Diamond v. Diehr*, 450 U.S. 175, 187 (1981) (subject matter not precluded from patent protection merely because it implements a "mathematical formula, computer program, or digital computer"). Although patent law confers broad protection, it is not considered flawless. See GEMIGNANI, *supra*, at 100. It is a "questionable means of protecting software" because it is expensive, time consuming to litigate, and difficult to prove non-obviousness." Andrew G. Rodau, *Protecting Computer Software: After Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240 (3d Cir. 1983), *Does Copyright Provide the Best Protection?*, 57 TEMP. L.Q. 527, 551-52 (1984).

Moreover, the restrictions on patent applications make it difficult for a software package to secure patent protection. See GEMIGNANI, *supra*, at 99-101. However, while an abstract idea or principle is not patentable, a process or device that uses it may be within the ambit of patent protection. "If a court determines that a program is identical with the algorithm it contains, then the court will necessarily hold against patentability." *Id.* at 101. Alternatively, if the program is simply "a useful process that utilizes the algorithm without preempting it, and the program clears the hurdles of novelty and unobviousness as well, the

to which elements are unprotectable as "ideas" and which elements are protectable as "expressions."¹⁰

This Note will examine the "idea-expression" dichotomy of copyright law as it pertains to computer software. Part One will discuss the historical development of software copyright law applying copyright infringement standards to computer software. Part Two will outline the steps involved in the development of a computer program. Part Three will discuss how the "idea-expression" distinction has developed through case law and the difficulty encountered by the judiciary in the software field. Part Four will present the Second Circuit's most recent attempt to enunciate a new standard in this area and will discuss the limitations of this stan-

court will find it patentable." *Id.*

Trade secrecy and contract law are the industry's most widely used methods of protection. See Friedman, *supra* note 5, at 32; RAYMOND T. NIMMER, *THE LAW OF COMPUTER TECHNOLOGY* ¶ 3.01, at 3-2 (1985). Trade secrets do not require registration and protection is immediately available for a program's logic and design including techniques, methods, processes, as well as the actual "source code" and "object code." See Friedman, *supra*, note 5, at 32; Anderson L. Baldy III, Note, *Computer Copyright: An Emerging Form of Protection for Object Code Software After Apple v. Franklin*, 5 *COMPUTER/L.J.* 233, 240 (1984). The legally protected interest in software is defined in the Restatement (First) of Torts:

[A trade secret consists of] information which is used in one's business and which gives him an opportunity to obtain an advantage over competitors who do not know or use it A substantial element of secrecy must exist, so that, by use of improper means, there would be difficulties in acquiring information Protection is not based on a policy of rewarding or otherwise encouraging the development of secret processes or devices. The protection is merely against breach of faith and reprehensible means of learning another's secret.

NIMMER, *supra*, ¶ 3.02, at 3-3 (quoting RESTATEMENT (FIRST) OF TORTS § 757 cmt. b (1939)) (brackets in original). In order for one to possess a protected trade secret, there must be expectations of secrecy between the creator and the potential misappropriator of novel and valuable information. *Id.* at 3-4. This requirement leads to several problems inherent in trade secret protection, particularly for mass-marketed software. See Peter S. Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 *STAN. L. REV.* 1046, 1077-78 (1989) (discussing difficulties in establishing and enforcing trade secret protection programs). For example, there are no judicially enforceable confidentiality restraints between the parties. See NIMMER, *supra*, ¶ 3.01, at 3-2. In addition, since trade secret protection is created by state law, it varies from state to state; thus, the level of protection also differs from one locale to the next. See Friedman, *supra* note 5, at 32; GEMIGNANI, *supra*, at 113.

10. See *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1235 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987). The distinction between "idea" and "expression" is elusive, and it will inevitably be decided on an ad hoc basis. *Id.* Judge Learned Hand stated that "[n]obody has ever been able to fix that boundary, and nobody ever can." *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930), *cert. denied*, 282 U.S. 902 (1931). Unfortunately, the CONTU report and the 1980 amendments to the Copyright Act did not explicitly state which elements of a program were protected and which were not. See Friedman, *supra* note 5, at 6-7.

dard. Finally, Part V will offer an alternative approach to provide copyright protection for computer software in light of existing policy considerations and the economic nature of the software industry.

I. SOFTWARE COPYRIGHT LAW

United States copyright law is founded on the constitutional power granted to Congress to "promote the Progress of Science and useful Arts."¹¹ Congress exercised this power by enacting the Copyright Act of 1976,¹² which declares that "copyright protection subsists . . . in original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device."¹³ Congress initially intended to afford copyright protection to computer programs.¹⁴ This intention was reaffirmed and clarified by the 1980 amendments to the Copyright Act,¹⁵ which classify computer pro-

11. U.S. CONST. art. I, § 8, cl. 8.

12. Act of October 19, 1976, Pub. L. No. 94-553, 99 Stat. 2541 (1976) (codified as amended at 17 U.S.C. §§ 101-108); see Friedman, *supra* note 5, at 5. The first copyright law was enacted in 1790, but the last thorough revision was passed in 1976. *Id.* (citation omitted); M. Kramer Mfg. Co. v. Andrews, 783 F.2d 421, 432 (4th Cir. 1986) (discussing development of copyright law).

"In attempting to fulfill its constitutional mandate . . . Congress has created a balance between the artist's right to control the work during the term of the copyright protection and the public's need for access to creative works." *Stewart v. Abend*, 495 U.S. 207, 209 (1990). The task of defining the scope of copyright protection under the Constitution requires Congress to balance "the interests of the authors and inventors in the control and exploitation of their writings and discoveries on the one hand, and society's competing interest in the free flow of ideas, information and commerce on the other hand." *Sony Corp. of Am. v. Universal City Studios, Inc.*, 464 U.S. 417, 429 (1984).

13. 17 U.S.C. § 102(a) (1988).

14. See H.R. REP. No. 1476, 94th Cong., 2d Sess. 51 (1976), reprinted in 1976 U.S.C.C.A.N. 5659, 5664. The House Report issued with the 1976 Copyright Act stated:

The history of copyright law has been one of gradual expansion in the types of works accorded protection In some of these cases the new expressive forms—electronic music, filmstrips, and computer programs, for example—could be regarded as an extension of copyrightable subject matter Congress had already intended to protect, and were thus considered copyrightable from the outset without the need of new legislation.

Id.

15. See Act of Dec. 12, 1980, Pub. L. No. 96-517, 94 Stat. 3015, 3028 (codified at 17 U.S.C. §§ 101, 117 (1988)) [hereinafter Act of Dec. 12, 1980]. CONTU's charter provided for a three year review period commencing in 1975. See CONTU REPORT, *supra* note 8, at 1. Thus, its recommendations were not compiled until after the enactment of the Copyright Act of 1976. *Id.*

grams as protectable "literary works."¹⁶

Copyright protection is only available for a work's "expression,"—not its "idea."¹⁷ Although this "idea-expression" dichotomy has long been recognized by the courts,¹⁸ determining the ex-

CONTU made a number of suggestions in the area of computer software:

The new copyright law should be amended 1) to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright; 2) to apply to all computer uses of copyrighted programs by the deletion of the present Section 117; and 3) to assure that rightful possessors of copies of computer programs can use or adapt these copies for their own use.

Id. at 2. The recommendations were intended to achieve the following goals:

- (1) Copyright should proscribe the unauthorized copying of these works.
- (2) Copyright should in no way inhibit the rightful use of these works.
- (3) Copyright should not block the development and dissemination of these works.
- (4) Copyright should not grant anyone more economic power than is necessary to achieve the incentive to create.

Id. at 29. The recommendations, however, were not unanimously supported by all members of CONTU itself. *Id.* at 2. Commissioner Hersey dissented, arguing that "copyright protection does not extend to a computer program in the form in which it is capable of being used to control computer operations." *Id.* The recommendations of CONTU were nevertheless adopted by Congress in the 1980 amendments to the Copyright Act. *See* Act of Dec. 12, 1980, *supra*, at 3028.

16. *See* 17 U.S.C. § 101 (1988). " 'Literary works' are works other than audiovisual work, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, . . . film, tapes, disks, or cards, in which they are embodied." *Id.* "[L]iterary works' include 'computer programs to the extent that they incorporate authorship in the programmer's expression of original ideas . . . [not] the ideas themselves.'" H.R. REP. NO. 1476, *supra* note 14, at 54, *reprinted in* 1976 U.S.C.C.A.N. at 5667. The 'literary works' classification is "one of seven copyrightable categories." *See* Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1249 (3d Cir. 1983), *cert. dismissed*, 464 U.S. 1033 (1984). "[A] computer program, whether in object code or source code, is a 'literary work' and is protected from unauthorized copying." *Id.*; *see also* 1 MELVILLE B. NIMMER & DAVID NIMMER, NIMMER ON COPYRIGHT § 2.04(c), at 2-46.2 to 2-46.3 (1991) (written computer programs are copyrightable as literary works). Additionally, case law has held that computer programs are copyrightable, under a separate category, as "original works of authorship in any tangible medium of expression." Williams Elecs., Inc. v. Artic Int'l, Inc. 685 F.2d 870, 873 (3d Cir. 1982) (quoting 17 U.S.C. § 102 (1976)).

17. *See* Baker v. Selden, 101 U.S. (11 Otto) 99, 103-04 (1879). "Idea" has been defined, *inter alia*, as "[a]ny conception existing in the mind as a result of mental understanding, awareness, or activity; . . . a thought, conception, or notion; . . . an impression." THE RANDOM HOUSE DICTIONARY OF THE ENGLISH LANGUAGE 949 (2d ed. 1987) [hereinafter RANDOM HOUSE DICTIONARY]. 'Expression' refers to the physical embodiment ideas." Roth Greeting Cards v. United Card Co., 429 F.2d 1106, 1109 (9th Cir. 1970); *see also* Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc., 797 F.2d 1222, 1234 (3d Cir. 1986) (only "expressions of ideas" protected), *cert. denied*, 479 U.S. 1031 (1987); Computer Assocs. Int'l, Inc. v. Altai, Inc., 775 F. Supp. 544, 558 (E.D.N.Y. 1991) (same), *aff'd* 1992 WL 372273 (2d Cir. 1992). *See generally* 1 NIMMER & NIMMER, *supra* note 16, at § 2.03[D] (only expression protected).

18. *See, e.g.,* Mazer v. Stein, 347 U.S. 201, 218 (1954) (discussing idea-expression "dichotomy"); Baker v. Selden, 101 U.S. (11 Otto) 99, 105 (1879) (same).

act point at which an "idea" is copyrightable as "expression" has always been problematic,¹⁹ particularly in the area of computer software.²⁰ A computer program is a "true hybrid" in terms of copyright law.²¹ While it is first created in written form as a series of commands,²² its practical value lies not in the writing itself, but in the accomplishment of its intended function.²³ One might view the written form as the "expression" and the functional form as the "idea," but some courts and commentators have criticized this approach as too simplistic in the area of computer software.²⁴

19. See *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930), *cert. denied*, 282 U.S. 902 (1931). Judge Learned Hand, a copyright authority, stated:

It is of course essential to any protection of literary property, whether at common-law or under the statute, that the right cannot be limited literally to the text, else a plagiarist would escape by immaterial variations. That has never been the law, but, as soon as literal appropriation ceases to be the test, the whole matter is necessarily at large, so that, as was recently well said by a distinguished judge, the decisions cannot help much in a new case . . . Nobody has ever been able to fix that boundary [between "idea" and "expression"], and nobody ever can.

Id.

20. See Gregory J. Maier, *Software Protection—Integrating Patent, Copyright and Trade Secret Law*, 69 J. PAT. & TRADEMARK OFF. SOC'Y, 151, 151 (1987). "It is the hybrid nature of software that causes its failure to fit neatly into any one existing category of intellectual property, resulting in seemingly endless confusion as to how it may best be protected." *Id.*

21. *Id.*

22. See *id.*

23. See *id.* at 151. "[F]unctionality . . . clearly distinguishes [software] from ordinary writings." *Id.* See generally Root, *supra* note 2, at 1266-68 (describing development and usefulness of operating and application programs).

24. See *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1236 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987). The court stated that "the purpose or function of a utilitarian work would be the work's idea, and everything that is not necessary to that purpose or function would be part of the expression of the idea." *Id.* The court further noted that when several ways exist to accomplish a desired result, the particular way selected is not necessary to the result, therefore, it is expression, not idea. *Id.* Limiting copyright protection to a program's literal elements was rejected. *Id.* at 1238. Further, the court was unconvinced that development in computer technology was significantly unlike that in "other areas of science or the arts." *Id.*

However, the *Whelan* test for determining copyrightability of computer programs has been sharply criticized by other courts. See *Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 775 F. Supp. 544, 558 (E.D.N.Y. 1991), *aff'd* 1992 WL 372273 (2d Cir. 1992). In *Altai*, the court found the standard measure, "similarity," totally "inadequate" when applied to computer programs since they were designed for application to artistic and literary works—not utilitarian works. *Id.* It was this "inadequacy," the court believed, that spurred the *Whelan* court "to set forth what now seems to be a *simplistic* test for similarity between computer programs." *Id.* (emphasis added).

Commentators have also criticized the *Whelan* test. See 3 NIMMER & NIMMER, *supra* note 16, § 13.03[F], at 13-62.34 (1991). "The crucial flaw in . . . [the *Whelan* test] is that it assumes that only one 'idea,' in copyright law terms, underlies any computer program, and

In a copyright infringement action, a plaintiff must establish ownership of a valid copyright and that the defendant copied the copyrighted work.²⁵ Copyright ownership is easier to establish than copying;²⁶ however, a plaintiff may prove copying circumstantially.²⁷ In order to do so, a plaintiff must show that the infringing party had "access" to the program²⁸ and that the infringed work is "substantially similar" to the copyrighted work.²⁹ If the material

that once a separable idea can be identified, everything else must be expression." *Id.*; see also Jack E. Brown, *The Current Status of Copyright Protection for Computer Software and Some Patent Protection Parallels*, 6 COMPUTER L. & PRAC. 170, 172 (1990) (critiquing *Whelan*).

25. See, e.g., *Feist Publications, Inc. v. Rural Tel. Serv. Co.*, 111 S. Ct. 1282, 1292 (1991); see also *Whelan*, 797 F.2d at 1231 (copyright ownership and actual copying must be shown); *Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp.*, 562 F.2d 1157, 1162 (9th Cir. 1977) (same); *Reyher v. Children's Television Workshop*, 533 F.2d 87, 90 (2d Cir.), *cert. denied*, 429 U.S. 980 (1976) (same).

26. See 17 U.S.C. § 408(a) (1988). Official copyright registration is not required in order for protection to exist. See *id.* Registration may occur at "[a]ny time during the subsistence of copyright in any published or unpublished work." *Id.* A person claiming copyright infringement may not, however, initiate an action until such registration has occurred. *Id.* at § 411(a) (1988). Registration "constitute[s] *prima facie* evidence of the validity of the copyright." *Id.* § 410(c) (1988) (emphasis added). See generally 3 NIMMER & NIMMER, *supra* note 16, § 13.01[A], at 13-4 to 13-8 (discussing copyright ownership).

On the other hand, direct evidence of copying is rarely available. See *Whelan*, 797 F.2d at 1231; *Roth Greeting Cards v. United Card Co.*, 429 F.2d 1106, 1110 (9th Cir. 1970).

27. See *Ferguson v. National Broadcasting Co.*, 584 F.2d 111, 113 (5th Cir. 1978).

28. See 3 NIMMER & NIMMER, *supra* note 16, § 13.02[A]. Historically, courts have defined "access" as the actual viewing of another's work. See *Schwarz v. Universal Pictures Co.*, 85 F. Supp. 270, 271 (S.D. Cal. 1945). Recently, courts have broadened this definition to "merely mean[] an opportunity to view the protected material." See *Robert R. Jones Assocs., Inc. v. Nino Homes*, 858 F.2d 274, 277 (6th Cir. 1988) (emphasis added).

29. See *Whelan*, 797 F.2d at 1231-32 (citations omitted); *Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 775 F. Supp. 544, 557 (E.D.N.Y. 1991), *aff'd*, 1992 WL 372273 (2d Cir. 1992).

The Second Circuit, in *Arnstein v. Porter*, 154 F.2d 464 (2d Cir. 1946), provided a test to determine if substantial similarity exists:

If there is evidence of access and similarities exist, then the trier of the facts must determine whether the similarities are sufficient to prove copying. On this issue, analysis ("dissection") is relevant, and the testimony of experts may be received to aid the trier of facts

If copying is established, then only does there arise the second issue, that of illicit copying On that issue . . . the test is the response of the ordinary lay hearer; accordingly, on that issue, "dissection" and expert testimony are irrelevant.

Id. at 468.

This is known as the *Arnstein* bifurcated test. See *Root*, *supra* note 2, at 1278-79 n. 99. The first step, the "extrinsic" test, entails the use of expert testimony to determine whether the defendant has copied the work. See *Krofft*, 562 F.2d at 1164. The second step, the "intrinsic" test, requires the observation of a lay observer to assess whether the two works are substantially similar. *Id.* Generally, this test has been used in cases involving materials such as "novels, plays and paintings." *Whelan*, 797 F.2d at 1232.

constitutes an "idea," however, the "substantial similarity" test becomes irrelevant because "ideas" are not entitled to copyright protection.³⁰

II. COMPUTER PROGRAM DEVELOPMENT

A computer program is the "set of statements or instructions to be used directly or indirectly in a computer to bring about a certain result."³¹ In order to create a program, a programmer must work in stages.³² First, the ultimate function or problem to be solved must be identified.³³ For example, a programmer may be asked to create a record-keeping program. Before such a program can be written, additional background information is needed, such as order processing, billing procedures, inventory requirements, and other characteristics of the particular trade.³⁴ As this information is gathered, the programmer moves into the second phase of outlining or flowcharting solutions to the problem.³⁵ Some flowcharts may represent minute processes that will be incorporated into the final program.³⁶ The interaction of the various algorithms³⁷ encompassed in the flowcharts is then analyzed and ar-

The second prong of the *Arnstein* bifurcated test has been rejected in computer software cases because the ordinary observer does not have the requisite scientific knowledge to determine how much, if any, of a program has been copied. *See* Friedman, *supra* note 5, at 8. Additionally, the most copyrightable expression, the machine language code, is "incomprehensible" even to computer experts. *Id.* at 8-9.

30. *See supra* note 17 and accompanying text; *see also* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 53 (D. Mass. 1990). Congress has mandated that the courts create a standard to determine the boundary between "idea" and "expression." *See id.*; *see also* Copyrights On Software, N.Y. TIMES, May 11, 1987, at D2 (noting that infringement claims decided on whether item in dispute is "idea" or "expression of idea").

31. 17 U.S.C. § 101 (1988).

32. *See* GEMIGNANI, *supra* note 9, at 80; Menell, *supra* note 9, at 1051; Friedman, *supra* note 5, at 4.

33. *See* GEMIGNANI, *supra* note 9, at 80; Menell, *supra* note 9, at 1052.

34. *See* Menell, *supra* note 9, at 1052. "[A] major aspect of the task definition process is understanding the users and determining how best to serve their needs." *Id.*

35. *See* Data Cash Sys., Inc. v. JS&A Group, Inc., 628 F.2d 1038, 1040 (7th Cir. 1980); Friedman, *supra* note 5, at 4. A flowchart itself may be copyrightable depending on its degree of detail. *See* NIMMER, *supra* note 9, ¶ 1.03[3], at 1-15 to 1-16. While a very simple flowchart would normally not qualify for copyright protection, a novel and detailed one may receive protection. *Id.* at 1-16. Flowchart is defined as "a graphic representation, using symbols interconnected with lines, of the successive steps in a procedure or system." RANDOM HOUSE DICTIONARY, *supra* note 17, at 738.

36. *See* Susan A. Dunn, Note, *Defining the Scope of Copyright Protection for Computer Software*, 38 STAN. L. REV. 497, 500-01 (1986). In the computer industry, the minute processes are often called subroutines or modules. *Id.*

37. The underlying processes used by a program are called algorithms. *See* Michael S. Keplinger, *Computer Software—Its Nature and Its Protection*, 30 EMORY L.J. 484, 484-85

ranged in the most efficient manner.³⁸ The programmer must then determine what input is necessary, in what order it should be input, and how it will be combined with other data.³⁹ Finally, the actual programming begins.⁴⁰ All steps in the process must be programmed into a language that the computer can understand,⁴¹ such as the familiar computer languages FORTRAN⁴² and BASIC.⁴³ Once written, such programs are in what is referred to as

(1981). An algorithm is the underlying process used by a program and is defined as a "fixed step-by-step procedure for accomplishing a given result," *Diamond v. Diehr*, 450 U.S. 175, 186 n.9 (1981) (citations omitted), or "[a] defined process or set of rules that leads and assures development of a desired output from a given input," *id.* Although algorithms must be developed by humans, computers can execute them faster and more accurately. See *Whelan Assocs., Inc. v. Jaslow Dental Lab., Inc.*, 797 F.2d 1222, 1229 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987) (citation omitted).

38. See Menell, *supra* note 9, at 1085. Efficiency relates to fast execution speed, minimal memory utilization and compatibility with data storage devices. *Id.* "While efficiency measures may vary across programs depending on the particular programming objectives, the range of efficiency goals is clear: faster processing speed, good programming practice (as a means of minimizing debugging problems), efficient memory capacity utilization, and rapid, accurate information transmission across interfaces." *Id.* Programs must be efficient to compete in the marketplace. See *Whelan*, 797 F.2d at 1230.

39. See *Whelan*, 797 F.2d at 1230. Such information may be organized in various ways, some more efficient than others and some more "quirky" than others, all of which contribute to the desirability of using a program. *Id.*; see also, JOHN C. LAUTSCH, AMERICAN STANDARD HANDBOOK OF SOFTWARE LAW § 3.2, at 28 (describing pre-programming analysis).

40. See LAUTSCH, *supra* note 39, § 3.2, at 28. Writing the software code is easy compared to the preparatory stages of outlining and flowcharting. See Friedman, *supra* note 5, at 16. "[A]mong the more significant costs in computer programming are those attributable to developing the structure and logic of the program." *Whelan*, 797 F.2d at 1237.

41. See Note, *Copyright Protection of Computer Program Code*, 96 HARV. L. REV. 1723, 1724-25 (1983). There exists two types of computer language, high level and low level. *Id.* Low level language includes "machine language" that is understood by the computer but not by humans. See *infra* notes 42-43, 46.

In order for humans to understand program form, high level languages were created. See Note, *supra*, at 1725. A "high level" computer language which employs English-like words is used by the programmer. See M. Margaret McKeown & Gregory J. Wrenn, *The Stakes on Secrecy Are Rising*, NAT'L L.J., Feb. 24, 1992, at 27. A second high level language, "assembly language," which is a series of alpha-numeric labels, is used to translate those commands into machine understandable code. See Note, *supra*, at 1725. For purposes of this Note, the "assembler" step was not presented in the text because it is not relevant to the analysis.

42. See Note, *supra* note 41, at 1725. FORTRAN is a "high level programming language used mainly for solving problems in science and engineering." RANDOM HOUSE DICTIONARY, *supra* note 17, at 754. The word "FORTRAN" stands for "formula translation." See *id.*

43. See STEVEN J. MANDELL, COMPUTERS AND DATA PROCESSING: CONCEPTS AND APPLICATIONS WITH BASIC 257 (2d ed. 1982). BASIC is "a widely adopted programming language that uses English words, punctuation marks, and algebraic notation to facilitate communication between the operator or lay user and the computer." RANDOM HOUSE DICTIONARY, *supra* note 17, at 173. "BASIC" is an acronym for "Beginner's All Purpose Symbolic Instruction

"source code."⁴⁴ The program is then translated into "object code" or binary code, which is a series of "0"s and "1"s⁴⁵ that actually instruct the computer.⁴⁶

III. JUDICIAL DECISIONS

The difficulty for the courts lies both in determining at what stage in the development of a computer program an "idea" becomes an "expression," and to what extent such expression is entitled to copyright protection.⁴⁷ The first generation of cases focused on the literal copying of the program's "source" or "object" code.⁴⁸ In *Stern Electronics, Inc. v. Kaufman*,⁴⁹ the Second Circuit held that a program's "source code" was copyrightable.⁵⁰ Subsequently, in *Williams Electronics, Inc. v. Artic International, Inc.*,⁵¹ the Third Circuit extended copyright protection to a program's "object code,"⁵² holding that such protection was mandated by statute.⁵³

Code." *See id.*

44. *See* RANDOM HOUSE DICTIONARY, *supra* note 17, at 173. Source code is a series of alphanumeric symbols and commands comprising the instructions to the computer as entered by the programmer. *Id.*; *see* NIMMER, *supra* note 9, ¶ 1.03[2], at 1-13. "The source program may then be transferred to some medium, such as cards or paper tape, for entry into the computer, or it may be entered directly through a terminal." GEMIGNANI, *supra* note 9, at 80.

45. *See* Menell, *supra* note 9, at 1051 n.27. An assembler or compiler translates the source code into machine readable "object code." *Id.*

46. *See* LAUTSCH, *supra* note 39, § 3.5, at 35. Source code is not capable of controlling the computer directly. *See* NIMMER, *supra* note 9, ¶ 1.03[2], at 1-13. It must be converted into electromagnetic impulses that are machine readable. *Id.*; *see also* Note, *supra* note 41, at 1724-25 (discussing process of code conversion).

47. *See* GEMIGNANI, *supra* note 9, at 80-81; *see also supra* note 17 and accompanying text.

48. *See, e.g., Apple Computer, Inc. v. Franklin Computer Corp.*, 714 F.2d 1240, 1249 (3d Cir. 1983) (stating object code and source code protected from copying by copyright law), *cert. denied*, 464 U.S. 1033 (1984); *Data Cash Sys., Inc. v. JS&A Group, Inc.*, 628 F.2d 1038, 1040-41 (7th Cir. 1980) (discussing copyright protection of coding in chips). *See generally* Menell, *supra* note 9, at 1073 (discussing first generation cases).

49. 669 F.2d 852 (2d Cir. 1982).

50. *Id.* at 855 n.3. "Written computer programs are copyrightable as literary works." *Id.* (citation omitted).

51. 685 F.2d 870 (3d Cir. 1982).

52. *Id.* at 876-77. The defendant argued that "object code" was not protected because it could not be understood by human beings. *See id.*

He argued that a human must comprehend the medium in order for copyright protection to exist. *Id.* Citing the Copyright Act, the court rejected this argument stating that protection extends to any work that "is fixed 'by any method now known or later developed, and from which work can be perceived . . . with the aid of a machine or device.'" *Id.* at 877 (citing 17 U.S.C. § 101 (1988)).

53. *Williams*, 685 F.2d at 873-74; *see Stern*, 669 F.2d at 855-56; *Apple Computer, Inc.*

Both courts reasoned that a program's literal code is entitled to copyright protection because it is an "original work[] of authorship fixed in a tangible medium."⁵⁴ Lower courts as well as other circuit courts have uniformly embraced the *Stern Electronics* and *Williams Electronics* holdings.⁵⁵

The second generation of cases involve the more difficult problem⁵⁶ of infringement of the "nonliteral" aspects of a computer program—duplication of a program's organization and structure without copying the actual source or object code.⁵⁷ In what is considered a landmark case, the United States Court of Appeals for the Third Circuit, in *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc.*,⁵⁸ attempted to clarify the distinction between "idea" and "expression."⁵⁹ According to the *Whelan* court, "*the purpose or function of a utilitarian work would be the work's idea, and everything that is not necessary to that purpose or function would be part of the expression of the idea.*"⁶⁰ In other words, if there are a number of alternative ways to perform a particular function, then each is an "expression."⁶¹ The court further stated, however, that no protection is afforded to "*scenes a faire*,"⁶² which are "incidents, characters or settings which are as a practical matter indispensable, or at least standard, in the treatment of a given topic."⁶³ The underlying rationale for the *scenes a faire* doctrine is that if only a limited number of ways to express

v. Franklin Computer Corp., 714 F.2d 1240, 1247 (3d Cir. 1983), cert. dismissed, 464 U.S. 1033 (1984).

54. *Williams*, 685 F.2d at 873-74 (quoting 17 U.S.C. § 102(a) (1988)).

55. See, e.g., *Atari Games Corp. v. Oman*, 888 F.2d 878, 882 (D.C. Cir. 1989); *Red Baron-Franklin Park, Inc. v. Taito Corp.*, 883 F.2d 275, 278 (4th Cir. 1989), cert. denied, 493 U.S. 1058 (1990); *United States v. O'Reilly*, 794 F.2d 613, 614-15 (11th Cir. 1986); *M. Kramer Mfg. Co. v. Andrews*, 783 F.2d 421, 434-35 (4th Cir. 1986); *Videotronics, Inc. v. Bend Elecs.*, 564 F. Supp. 1471, 1477 (D. Nev. 1983); *Midway Mfg. Co. v. Strohon*, 564 F. Supp. 741, 746 (N.D. Ill. 1983); *Nintendo of Am., Inc. v. Elcon Indus., Inc.* 564 F. Supp. 937, 943 (E.D. Mich. 1982); *In re C Tek Software, Inc.*, 127 B.R. 501, 506 (Bankr. D.N.H. 1991).

56. See Menell, *supra* note 9, at 1073-74.

57. *Id.* at 1074.

58. 797 F.2d 1222 (3d Cir. 1986), cert. denied, 479 U.S. 1031 (1987).

59. *Id.* at 1235-42.

60. *Id.* at 1236.

61. *Id.*

62. *Id.* (citations omitted). *Scenes a faire* "receive protection only from virtually identical copying." *Atari, Inc. v. North Am. Philips Consumer Elecs. Corp.*, 672 F.2d 607, 617 (7th Cir.), cert. denied, 459 U.S. 880 (1982); see also *Frybarger v. International Business Machs. Corp.*, 812 F.2d 525, 530 (9th Cir. 1987) (noting that indispensable expressions receive protection only from identical copying).

63. *Whelan*, 797 F.2d at 1236 (citation omitted).

an idea exists, then the "idea" of the work can only be accomplished by employing one of these methods;⁶⁴ thus, copyright protection would guarantee a virtual monopoly over a particular necessary activity,⁶⁵ contrary to the purposes of copyright law.

Applying this analysis to the facts of its case, the *Whelan* court found that "the structure of the program was not essential to [the] task" performed and, therefore, was an "expression," entitled to copyright protection.⁶⁶ In reaching its decision, the Third Circuit adopted the district court's definition of "expression:" "The 'expression of an idea' in a software computer program is the manner in which the program operates, controls and regulates the computer in receiving, assembling, calculating, retaining, correlating, and producing useful information either on a screen, print-out or by audio communication."⁶⁷

This definition has led to inconsistent decisions among lower courts because some have interpreted it to afford extremely broad protection and others very limited protection.⁶⁸ Moreover, the other circuit courts have not uniformly embraced the *Whelan* test.⁶⁹

In *Plains Cotton Cooperative, Association v. Goodpasture Computer Service, Inc.*,⁷⁰ the Fifth Circuit rejected *Whelan* stating that copyright protection should not extend to the structure of a computer program because the structure represents "ideas" rather than "expression."⁷¹ However, a closer examination of the *Plains*

64. See *id.*; *Walker v. Time Life Films, Inc.*, 615 F. Supp. 430, 436 (S.D.N.Y. 1985), *aff'd*, 784 F.2d 44 (2d Cir.), *cert. denied*, 476 U.S. 1159 (1986); see also *Narell v. Freeman*, 872 F.2d 907, 911 (9th Cir. 1989) (stating that ordinary phrases not entitled to copyright protection); *Atari Games Corp. v. Oman*, 888 F.2d 878, 886 (D.C. Cir. 1989) (explaining that *scenes a faire* refer to stereotyped expressions).

65. See *Landsberg v. Scrabble Crossword Game Players, Inc.*, 736 F.2d 485, 489 (9th Cir.), *cert. denied*, 469 U.S. 1037 (1984).

66. See *Whelan*, 797 F.2d at 1238-39. "[T]here are other programs on the market . . . that perform the same functions but have different structures and designs." *Id.* at 1238.

67. *Id.* at 1239 (citing *Whelan*, 609 F. Supp. at 1320).

68. Compare *Broderbund Software, Inc. v. Unison World, Inc.*, 648 F. Supp. 1127, 1135 (N.D. Cal. 1986) (giving copyright protection to menu screens, input formats, and screen sequencing), with *Digital Communications Assocs., Inc. v. Softklone Distrib. Corp.*, 659 F. Supp. 449, 455 (N.D. Ga. 1987) (refusing to extend copyright protection to screen displays generated by program), and *Manufacturers Technologies, Inc. v. CAMS, Inc.*, 706 F. Supp. 984, 994-95 (D. Conn. 1989) (same).

69. See *supra* note 24 and accompanying text.

70. 807 F.2d 1256 (5th Cir.), *cert. denied*, 484 U.S. 821 (1987).

71. *Id.* at 1262. Citing Judge Higginbotham's opinion in *Synercom Technology, Inc. v. University Computing Co.*, 462 F. Supp. 1003 (N.D. Tex. 1978), the *Plains Cotton* court stated that a computer program's "input formats" were ideas, not expressions, and were not

Cotton decision reveals that it is actually reconcilable with *Whelan*. In *Plains Cotton*, the court determined that the structure of the program was dictated by the "externalities of the market,"⁷² implying that the "idea" was only expressible in a limited number of ways.⁷³ The structure of the program may therefore be seen as a "necessary incident," or *scenes a faire*, which is not an "expression," but merely an uncopyrightable "idea."⁷⁴ It is submitted that this conclusion is entirely consistent with the *Whelan* rule that when a computer program can be structured in several different ways to accomplish the same result, it is protected by copyright law.⁷⁵

IV. THE *COMPUTER ASSOCIATES* TEST

A more recent attack on *Whelan* was launched in *Computer Associates International, Inc. v. Altai, Inc.*,⁷⁶ in which the Second Circuit Court of Appeals rejected the *Whelan* test, calling it "flawed," "inadequate," and "outdated."⁷⁷ In addition to stating that the *Whelan* approach "relies too heavily on metaphysical distinctions and does not place enough emphasis on practical considerations,"⁷⁸ the court presented an alternative approach for determining computer software copyright infringement cases.⁷⁹ Borrowing from familiar copyright doctrines, the Second Circuit created a three-step test,⁸⁰ which, in essence, seems to merge the test for distinguishing between idea and expression with a method for ascertaining the "substantial similarity" of two computer pro-

entitled to copyright protection. *Id.* (citation omitted).

72. *Plains Cotton*, 807 F.2d at 1262. "The record supports the inference that market factors play a significant role in determining the sequence and organization of cotton marketing software, and we decline to hold that those patterns cannot constitute 'ideas' in a computer context." *Id.*; see *supra* notes 60-66 and accompanying text.

73. See *supra* notes 62-66 and accompanying text.

74. *Id.*

75. *Id.*

76. Nos. 762, 91-7893, 91-7935, 1992 WL 372273 (2d Cir. June 22, 1992).

77. *Id.* at *12. Judge Pratt, sitting by designation in the district court, also called the *Whelan* test "simplistic," "inadequate," and "inaccurate." See *Computer Assocs. Int'l, Inc. v. Altai, Inc.*, 775 F. Supp. 544, 558-59 (E.D.N.Y. 1991), *aff'd*, 1992 WL 372273 (2d Cir. 1992); Richard Raysman & Peter Brown, *Whelan Standards Attacked*, N.Y.L.J., Jan. 14, 1992, at 3.

78. *Computer Assocs.*, 1992 WL 372273, at *13.

79. *Id.* at *12-*20.

80. *Id.* at *12. The Second Circuit also cautioned other courts not to apply the test strictly in all circumstances. *Id.* If necessary, other courts should use a modified version of the test presented. *Id.*

grams. Step one of the test, referred to as the "abstractions test,"⁸¹ is used to determine which portions of the plaintiff's program are copyrightable expressions. The "abstractions test," as first enunciated by Judge Learned Hand, provides:

Upon any work . . . a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out. The last may perhaps be no more than the most general statement of what the [work] is about and at times might consist only of its title; but there is a point in this series of abstractions where they are no longer protected, since otherwise the [author] could prevent the use of his "ideas" to which, apart from their expression, his property is never extended.⁸²

The Second Circuit stated that, in the computer software context, this "abstractions test" would require a court to "dissect the allegedly copied program's structure and isolate each . . . abstraction contained within it."⁸³ For example, in the steps outlined in Part II of this Note, a court would analyze isolated steps of a program in reverse developmental order.⁸⁴ Possible abstractions would be the "object code," followed by the "source code," then the input formats, then the flowcharts or outlines, and finally the problem to be solved.⁸⁵

The second step is a "successive filtering method" that requires the court to screen a program's "components at each level of abstraction" to determine copyrightability of these components.⁸⁶ In so doing, a court must determine whether the inclusion of each component "was 'idea' or was dictated by considerations of efficiency, so as to be necessarily incidental to that idea; required by factors external to the program itself; or taken from the public domain and hence not protectable expression."⁸⁷ With this second step, the *Computer Associates* court reaffirmed the principle that

81. *Id.* at *12-*13.

82. *Nichols v. Universal Pictures Corp.*, 45 F.2d 119, 121 (2d Cir. 1930), *cert. denied*, 282 U.S. 902 (1931), *quoted in Computer Assocs.*, 775 F. Supp. at 560. The court, in *Computer Associates*, followed Professor Nimmer's suggestion that the abstractions test was the best method of determining substantial similarities in computer programs. *Id.* The Second Circuit was in "substantial agreement" with the district court's reasoning. *Computer Assocs.*, 1992 WL 372273, at *1.

83. *Computer Assocs.*, 1992 WL 372273, at *13.

84. *Id.*; see *supra* notes 31-46 and accompanying text.

85. See *supra* notes 31-46 and accompanying text.

86. *Computer Associates*, 1992 WL 372273, at *14-*17.

87. *Id.* at *14-*17.

scenes a faire, necessary incidents, and items in the public domain are not granted copyright protection.⁸⁸ Based on this approach, copyright protection would not be available for a computer program that is composed of existing techniques, even though taken as a whole the program may be novel.⁸⁹

The third step is "comparison."⁹⁰ Once a court eliminates the unprotectable components through the filtration process, "a core of protectable expression" remains.⁹¹ These "golden nuggets" are reviewed to determine whether "the defendant copied any aspect of this [remaining] protected expression," and to assess "the copied portion's relative importance with respect to the plaintiff's overall program."⁹²

It is submitted that although the court attempted to provide a "pragmatic" analysis, it failed to establish a truly useful approach, and instead placed additional unnecessary burdens on the lower courts. First, the filtration method proposed by the court requires the reverse engineering of a plaintiff's program.⁹³ This will necessarily increase the costs and time involved in software copyright litigation because experts must be employed to review the subject programs. The possibility of prohibitive legal and expert costs will likely result in foregone lawsuit opportunities, even though plaintiffs may have valid claims. Second, it is often unclear what is in the public domain and what is not, particularly in the dynamic field of computer software.⁹⁴ Therefore, a court may require additional expert advice to make such a determination, or it may even make a decision relying on mere speculation. In either case, consistency is unlikely. Finally, requiring courts to evaluate the relative importance of the copied portion of a program with the overall program places too much discretion with the court and will ultimately lead to further inconsistent decision-making. This will leave programmers uncertain as to which components of a program they

88. *Id.*

89. See Richard Raysman & Peter Brown, *Copyright Protection for Software Redefined*, N.Y.L.J., Jul. 14, 1992, at 3, 7.

90. *Computer Assocs.*, 1992 WL 372273, at *18.

91. *Id.*

92. *Id.*

93. *Id.* at *13. The court must examine the program "in a manner that resembles reverse engineering." *Id.*

94. See, e.g., Bill Machrone, *Roots: The Evolution of Innovation*, P.C. MAGAZINE, May 26, 1987, at 166, 169 (discussing the keystroke sequence of WordStar and how it became public domain).

are entitled to duplicate and which they are not.⁹⁵ It is submitted, therefore, that in its attempt to restrict copyright protection for computer software, the Second Circuit failed to confine copyright protection to the appropriate extent.

V. A PROPOSED ALTERNATIVE APPROACH

By providing a system of incentives, copyright protection⁹⁶ allows individual authors and developers to capitalize upon their creative labor.⁹⁷ The ultimate goal of copyright law, however, is to recognize and serve the public interest.⁹⁸ Indeed, Congress provides copyright protection "to serve the public welfare by encouraging authors to generate new ideas and disclose them to the public, being free to do so in any uniquely expressed way they may choose."⁹⁹ In defining copyright laws, Congress has the difficult task of balancing the interests of authors and developers without

95. See Jim Seymour, *Who Owns the Standards?*, P.C. MAGAZINE, May 26, 1987, at 174, 176. "In the meantime, software developers will have to consult a lawyer when designing their [programs]." *Id.*

96. See *Lotus Dev. Corp. v. Paperback Software Int'l*, 740 F. Supp. 37, 52-53 (D. Mass. 1990). "The economic philosophy behind the clause empowering Congress to grant patents and copyrights is the conviction that encouragement of individual effort by personal gain is the best way to advance public welfare through the talents of authors and inventors." *Mazer v. Stein*, 347 U.S. 201, 219 (1954); see also *United States v. Paramount Pictures, Inc.*, 334 U.S. 131, 158 (1948) (noting that reward to authors is inducement for them to release created products to the public), *aff'd*, 339 U.S. 974 (1950).

97. See *Fox Film Corp. v. Doyal*, 286 U.S. 123, 127 (1932).

The production to which the protection of copyright may be accorded is the property of the author and not of the United States. . . . [The copyright] confers upon the author after publication the exclusive right for a limited period to multiply and vend [the creation] and to engage in the other activities described by the statute in relation to the subject matter. . . . In creating this right, the Congress did not reserve to the United States any interest in the production itself, or in the copyright, or in the profits that may be derived from its use. . . . The owner of the copyright, if he pleases, may refrain from vending or licensing and content himself with simply exercising the right to exclude others from using his property.

Id. (citations omitted).

98. See *Sony Corp. v. Universal City Studios, Inc.*, 464 U.S. 417, 429 (1984); *Paramount Pictures*, 334 U.S. at 158; *Lotus*, 740 F. Supp. at 52-53.

The private reward to the author is not the goal of copyright law, it is merely an incentive, chosen by Congress, to achieve "the ultimate goal of copyright law—the advancement of public welfare." *Lotus*, 740 F. Supp. at 53. "The immediate effect of our copyright law is to secure a fair return for an 'author's' creative labor. But the ultimate aim is, by this incentive, to stimulate artistic creativity for the general public good." *Sony*, 464 U.S. at 432 (citations omitted).

99. *Lotus*, 740 F. Supp. at 52 (citing *Harper & Row, Publishers, Inc. v. Nation Enters.*, 471 U.S. 539 (1985)).

diminishing the free flow of ideas and information.¹⁰⁰ The Supreme Court has recognized this balancing effort, but has emphasized that, in deciding copyright cases, the policy of serving the public interest predominates.¹⁰¹

It is proposed that a program's "expression" should be limited to the literal elements of "source" and "object" codes, thereby providing a clear, predictable test for copyright infringement of software. Application of the existing tests may lead to overprotection to the detriment of innovation and public benefit.¹⁰² In the software industry, computer programs are built on pre-existing knowledge;¹⁰³ thus, improvements are incremental and founded on minor modifications to proven techniques.¹⁰⁴ Because the broad

100. See *supra* notes 11-13.

101. See *Sony*, 464 U.S. at 429 (citing *Paramount Pictures*, 334 U.S. at 158).

102. See *Root*, *supra* note 2, at 1293. "Computer programs are fundamentally different from most other literary works that produce technological growth in that the program itself both expresses the innovation and performs the new operation." *Id.* at 1292. In creating a program, a programmer must first recreate a copyrighted program's procedure if he is going to improve upon it. *Id.* at 1292-93. If broad protection is enforced a programmer will not be allowed to use existing knowledge and technology. *Id.* Providing such protection will create "strong monopolies" to those who first write programs, thereby "inhibit[ing] other creators from developing improved products." Menell, *supra* note 9, at 1047-48.

In addition, the user group will be less likely to purchase programs that require new training when they are familiar with existing programs. See *Synercom Technology, Inc. v. University Computing Co.*, 462 F. Supp. 1003, 1013 (N.D. Tex. 1978); Friedman, *supra* note 5, at 17. This is particularly true in the user interface area. See *Synercom*, 462 F. Supp. at 1013. In *Synercom*, the court emphasized the public benefit of allowing the use of a pre-existing "figure-H" automobile gear shift pattern. *Id.* Although the pattern was arbitrarily chosen, and could have been expressed in several other ways, the court found that uniform use of such a pattern was "socially desirable." *Id.* Such uniformity, contrasted with the ongoing creation of new patterns, avoids otherwise extensive and expensive training. *Id.* In applying this analogy to computer programs, the court said, "[a]dmittedly, there are many more possible choices of computer formats, and the decision among them more arbitrary, . . . [however] Synercom's argument that the order and sequence of data was the expression, not the idea, has been rejected." *Id.* (footnote omitted); see also Patricia Keefe, *Software Copyright a Mixed Bag*, *COMPUTERWORLD*, Feb. 12, 1990, at 35, 40 (user group seeks some level of compatibility among software packages, which would be threatened by broad copyright protection); Friedman, *supra* note 5, at 17 (potential user unlikely to purchase new software product if not compatible with market leader).

103. See *Root*, *supra* note 2, at 1291-92. Programmers are generally familiar with the function and design of the competitors' programs. See 3 *NIMMER & NIMMER*, *supra* note 16, § 13.03[F], at 13-26 -27. "User attachment to the particular way a popular program functions has spawned a segment of the software industry devoted almost exclusively to 'cloning' existing programs - independently producing less expensive look-alike and work-alike versions of popular programs." *Id.* at 13-78.27 n.272.

104. See Friedman, *supra* note 5, at 17; see also *Affiliated Hosp. Prods., Inc. v. Merdel Game Mfg. Co.*, 513 F.2d 1183, 1188 (2d Cir. 1975) (copyright infringement not found where "good faith attempt" to incrementally improve on existing knowledge).

protection granted under the existing tests will foreclose a great deal of this "cumulative innovation,"¹⁰⁵ the "stepping stone" innovation process, which is unique to the industry, will be less likely to proceed.¹⁰⁶ Consequently, such broad protection will force programmers to "start from scratch" every time an innovative program is considered,¹⁰⁷ a result antithetical to the very goal of the copyright laws.¹⁰⁸

Additionally, since computer technology is one of the most rapidly changing and developing industries,¹⁰⁹ current innovations may be obsolete in twelve months.¹¹⁰ In order for a programmer to copy another program's structure, considerable time and effort will be expended emulating the "ideas" behind the program.¹¹¹ As the

There are innumerable programs available that are based on improvements to existing products. See Machrone, *supra* note 94, at 166. "Users would be far worse off if they had to wait for conceptual breakthroughs or new paradigms instead of refinement." *Id.* Advancement that comes about through evolution, rather than revolution, should not be protected, and if such protection exists, the public will ultimately lose. *Id.* at 168.

105. See Root, *supra* note 2, at 1293. "[F]ew programmers have the capability to create totally new methods of operations, [however], there are many programmers who can mimic the pioneers and add improvements." *Id.* The costs to software developers may be prohibitive if they must consult an attorney every time they consider creating a new program. See Seymour, *supra* note 95, at 174, 176; see also Victoria Slind-Flor, *Lawyers, Programmers Interface*, NAT'L L.J., Mar. 16, 1992, at 3, 12 (stating that fear of cost of defending infringement suits chills programmer creativity); Paul Freiberger, *For This Suit, You Need a Program*, NEWSDAY, Feb. 18, 1992, at 27 (noting that lawsuits could prevent companies from developing applications programs).

106. See Root, *supra* note 2, at 1293. Every time a programmer intended to create a program, he would have to "reinvent the wheel." *Id.*; see also Machrone, *supra* note 94, at 168 (explaining that interfaces created through evolution belong in public domain and should not be protected).

107. See Friedman, *supra* note 5, at 17. "A programmer should . . . be free to study a piece of software and to incorporate any ideas he may glean from normal usage into his own work." *Id.* at 16. According to a 1989 survey, 80% of approximately 700 software developers were against infringement suits because of their negative impact on the industry. See Keefe, *supra* note 102, at 40.

108. See *supra* notes 97-101 and accompanying text (discussing public policies underlying copyright law).

109. See *supra* notes 1, 3 (discussing computer statistics).

110. See Robert N. Noyce, *Microelectronics*, 237 SCI. AM. 62, 68 (1977). "A year's advantage in introducing a new product or new process can give a company a 25% cost advantage over competing companies; conversely, a year's lag puts a company at a significant disadvantage with respect to its competitors." *Id.*

111. See James Chesser, Note, *Semiconductor Chip Protection: Changing Roles for Copyright and Competition*, 71 VA. L. REV. 249, 287-88 (1985).

Without Protection . . . there remains much financial incentive to innovate.

The lead-time gained by an innovator who is first to market a good product, creates special profits. Indeed, those who are first enjoy a form of product monopoly that lasts until they are copied. Moreover, copying is seldom immediate; generally,

Whelan court noted, "[o]ne cannot simply 'approximate' the entire copyrighted computer program and create a similar operative program without the expenditure of almost the same amount of time as the original programmer expended."¹¹² Thus, by the time a program's ideas are copied, reprogrammed, and manifested in a new form, it would already be obsolete because superior versions of existing programs will already be available.¹¹³ Obviously, it is unreasonable for a programmer to engage in such a fruitless activity. Instead, it would be more practical for a programmer to innovate and build on the existing technology.¹¹⁴

CONCLUSION

Limiting protection to the literal aspects of software would allow developers to reap the rewards of their creations and would provide an incentive for programmers to improve existing products or develop new programs in order to stay ahead of the competition. Society would thus benefit because competition would ensure continued technological growth in the computer software industry and the availability of the best possible software products. In addition, literal protection would provide a legal analysis that is both coherent and predictable. This results in more certainty not only for the courts, but also for the software industry and society as a whole.

Martin T. Hillery

the more sophisticated or involved a . . . design is, the longer it will take to copy. More fundamentally, it is competition that provides much of the impetus behind innovation.

Id. (footnotes omitted); see also *supra* note 102 and accompanying text.

112. *Whelan Assocs., Inc. v. Jaslow Dental Lab.*, 797 F.2d 1222, 1237 (3d Cir. 1986) (citation omitted), *cert. denied*, 479 U.S. 1031 (1987).

113. See *supra* note 110.

114. See Friedman, *supra* note 5, at 17.